Instrument Science Report ACS 2015-05

Basic Use of SExtractor Catalogs With TweakReg - II

Ray A. Lucas
May 27, 2015

ABSTRACT

*The **TweakReg** task in the **DrizzlePac** software package for aligning and drizzling images is an effective tool for the initial processing of HST images. However, the internal object-finding software which it uses to generate catalogs to be used for object matching and alignment, **ImageFindPars**, is based on the well-known stellar-oriented **daofind** routine, and though it is appropriate for most images, there are some for which the task is not well-suited, such as some extragalactic fields which have lots of galaxies but almost no stars, and in some cases, if taken over long time-baselines, even the few stars present may have moved due to proper motion, thereby making them unsuitable for use in image alignment. In that case, other object-detection programs such as SExtractor can be used to generate external catalogs of primarily galaxies which can then be fed to the **TweakReg** task. Even multiwavelength morphological differences may make some galaxies less suitable than others for use in image alignment and registration. The basic use of simple external SExtractor catalogs with the **DrizzlePac TweakReg** task is described here, using ACS/WFC data as an example. This includes updating geometric distortion files, initial drizzling to make cosmic ray-cleaned images, the set-up and use of required SExtractor files to generate SExtractor catalogs, basic manipulation of those catalogs, and their use with the **TweakReg** task in the **DrizzlePac** package. More options are explored for iteratively improving the astrometric solution than in Paper I (ACS ISR 2015-04) but this still represents an intermediate level of sophistication in methods. This paper introduces several methods but the ultimate best methods can vary with the nature of the objects and the data, etc. and are for the user to explore. Also, the use of crclean.fits images may be more suited to ACS/WFC and WFC3/UVIS than WFC3/IR.*

Introduction

Although the stellar-oriented, built-in object detection routine, **ImageFindPars**, in **DrizzlePac's TweakReg** task, can be quite useful for aligning images in most fields which have plenty of stars for alignment, there are some situations in which there are not many or any stars in the images to be aligned, and in which even those few stars may have moved due to proper motion if the images were taken over a long enough time-baseline, such as archival data from different observing programs on the same target separated by some number of years. Additionally, morphological differences in multiwavelength sets of images of galaxies may even make some galaxies less suitable than others as reference objects for registration and alignment. In this example, we are addressing the simplest case, merely using SExtractor software to generate very simple external catalogs to feed to the **TweakReg** task for alignment of images.

SExtractor, however, needs good, clean images to produce better results, but images, by default, usually have many cosmic rays in them which complicate the tasks of object detection, position measurement, and ultimately, image alignment. Although new developments with the default **ImageFindPars** task and **TweakReg** allow for use of the data quality (DQ) arrays to ignore or eliminate the use of objects affected by cosmic rays, SExtractor does not use the DQ array information, hence the need for first making the cr-clean.fits images.

In terms of ideal conditions for attacking this problem, if such images are available, it is always best to start by aligning and combining images from the same visit (the same guide star acquisition) and in the same filter, and then later expanding that to aligning and combining across visits in the same filter, and ultimately across both different filters and different visits. Therefore, in general, if designing an observing program, it is also a good idea to keep this in mind as an ideal set-up for aligning and drizzling images together. If you can have multiple images of similar exposure time in the same filter and same visit, then do that. It may not always be possible, but that is a good, ideal situation.

In this example, we have taken an even simpler approach, where all of the images in all of the filters were taken in one single visit. We have used the ACS SM4 ERO images of the galaxy cluster Abell 370 from HST program 11507 (PI: Noll). The data were taken in 2009, not long after SM4, and are comprised of 6 dithered ACS/WFC images in each of three filters, F475W, F625W, and F814W. A formatted version of the Phase II file may be seen at http://www.stsci.edu/hst/phase2-public/11507.pro via the standard program information web pages. We do not push the limits of everything we could try nor the limit of what could be achieved in this example, but we show enough to give an idea of how the solution can be improved.

Process

We will first (1) update the WCS and geometric distortion in each of the images - a process specific to ACS/WFC which currently has to be done manually, but which will be automated as part of the routine pipeline by late 2015 as part of OPUS 2015.3. Next, we will (2) make an initial run of **AstroDrizzle** on the same-filter, same-visit images to generate cosmic ray-cleaned images called *_crclean.fits which will be used in (3) generating the SExtractor catalogs. After generating the SExtractor catalogs from the crclean.fits images, we will (4) feed those catalogs to **TweakReg**, and (5) manipulate the input catalogs if needed/desired, iterating in each filter internally as needed without updating the headers until we have a solution which we deem more acceptable for each of the images in a given filter, and then moving on to the next filter, until we have a solution which seems reasonable for each filter. (Remember that in this case, all the data were taken within the same visit.) At that point, we will (6) re-run the **TweakReg** task for the images in each same-filter, same-visit set, this time updating the header information with the solution required to align all the images in a given filter for that visit.

Although perhaps not fully covered in this example since there are other examples regarding this, in typical usage and situations, the resulting per-filter-aligned images can then be (7) drizzled together again, and the drizzled output used as a reference image to be fed to the **DrizzlePac TweakBack** task to update the headers of all the images in all of the filters to that reference image, after which they can all be drizzled together on a per-filter basis with the resulting output also being aligned across both all visits and all filters.

We will show examples of task parameter files, of SExtractor set-up files' input settings and their content, of the various output catalogs and their content and format, and show some possibly-helpful means of manipulation of the output catalogs, etc. as well as giving some advice as to other things one may try in order to make the solution better since this paper is merely describing a general approach and not detailing all of the possibilities.

Step 1: Update the WCS Information for Geometric Distortion

This first step will only be required to be done manually until late 2015, when this part of the process is to be incorporated into the routine pipeline processing in OPUS 2015.3. Until that time, one must download new reference files for the IDCTAB, NPOLFILE, and D2IMFILE and use the **setval** task to first update the keyword values for the IDCTAB, NPOLFILE, and D2IMFILE keywords in the science image headers, and then use the task **updatewcs** to apply the corrections from the new files to the science

images. Bear in mind that if you update one of these files, you should update all three, as they are intended for use together as an ensemble. Throughout this example, it is assumed that the standard pipeline-calibrated, de-striped, CTE-corrected *_flc.fits images are being used, and those are the ones to which these geometric corrections are being applied.

This process is currently described in more detail on the ACS Distortion web page at http://www.stsci.edu/hst/acs/analysis/distortion and can briefly be summarized thus, as taken from the text of that page:

```
--> from astropy.io import fits
--> fits.setval(image_filename, 'IDCTAB',
value='/location/of/IDCfile/IDCFILEname.fits')

--> fits.setval(image_filename, 'NPOLFILE',
value='/location/of/NPOLfile/NPOLFILEname.fits')

--> fits.setval(image_filename, 'D2IMFILE',
value='/location/of/D2IMfile/D2IMFILEname.fits')

--> from stwcs import updatewcs
--> updatewcs.updatewcs('*flc.fits') # update the WCS in
the header.
```

Also, as per instructions on that page, be sure not to use the **updatewcs** task again as part of your subsequent **TweakReg** and **AstroDrizzle** runs as you try to find an optimal alignment since re-running the **updatewcs** task will over-write any previous corrections you have made with your initial run and subsequent iterations of **TweakReg**. Instead, use the *updatehdr* keyword with value set to "False" ("No") during your **TweakReg** iterations and then change it to "True" ("Yes") when you have found a solution you like and are re-running **TweakReg** for the last time.

Step 2: Initial AstroDrizzle Run to Make Cosmic-Ray-Cleaned Images

In this and subsequent sections on **AstroDrizzle** and **TweakReg**, you may run python commands or use the Teal interface to execute the tasks. These are described elsewhere, and we will merely describe the non-default parameter values which have been selected.

In order to make initial cosmic-ray-cleaned images to feed to SExtractor, we make a separate directory for all the images in a given filter for that visit and place the images for that filter into it. Then, we make use of the existing WCS information in the science

image headers to first combine the images on that same-visit, same-filter basis. Since we have 6 images in the same filter per visit, we use *combine_type = iminmed* in making the median image, and *driz_cr_corr = True*. If desired, a "final" drizzled science image (though substandard without any better alignment being applied) with an associated weight image of type IVM can be created using *driz_combine = True* and *final_wht_type = IVM*, as is needed for input to SExtractor, but for now we will ignore the use of weight maps with SExtractor in this basic example. The use of weight maps with SExtractor, a usage which tends to yield better positional accuracy, though not inherently difficult, is a more advanced topic which will be addressed further in, for example, Hubble Frontier Fields technical documentation. For now, for the main points being made in our basic example, we do not need an image of this sort, although use of weight images with SExtractor often, with iteration, ultimately tends to yield better positional accuracy. We will therefore in this example continue on with the use of the *_crclean.fits images which we have now created with our initial **AstroDrizzle** run, which has produced a single *_crclean.fits image for each single input *_flc.fits image, so that there is a one-to-one correspondence between these two image types. Appendix A lists our input parameters in detail for our initial run of **AstroDrizzle** to create these *_crclean.fits images.

Step 3: Run SExtractor to Produce Object Catalogs for Alignment

Here we assume that the reader has either already downloaded and installed SExtractor according to standard directions or will do so, and will refer as needed to further documentation referenced in this article. We have used SExtractor version 2.8.6 in this case. It handles multi-extension FITS files in a more user-friendly fashion than earlier versions in frequent use such as version 2.5.0. In SExtractor version 2.5.0, the software will run on multi-exension FITS images, but it runs on all extensions, and it is then necessary to make the extension number one of the outputs and to then filter out the catalogs generated for all other extensions except for the one(s) desired. In contrast, in SExtractor version 2.8.6, one may run the software on a single image extension at a time, but there is an offset which must be applied. Although in SExtractor version 2.5.0 output, the image extensions [1] and [4] correctly refer to the two science image extensions of primary concern for measuring and cataloging object positions, in SExtractor version 2.8.6, one must refer to those extensions as [0] and [3] respectively when running SExtractor on those two science image extensions.

When running SExtractor, 4 basic set-up files are involved: (1) **default.sex**, which is where parameters for minimum object size, background filters, detection threshold levels, photometric systems and zeropoints, *seeing_fwhm*, and *pixel_scale* (both only used for SExtractor's *class_star* galaxy-star discriminator), and other similar items fundamental to

the objects themselves are defined, (2) **default.conv**, which applies convolution filters of the user's choice (gaussian, Mexican hat, etc.) in object detection and characterization, (3) **default.nnw**, which defines a neural network for object classification, and (4) **default.param**, which defines the quantities to be output into the catalog from the SExtractor run. These input files are automatically called when SExtractor is run. Examples of these 4 input files are shown in Apendices B, C, D, and E. In our usage here, we have used mostly default parameters in the creation of these simple, basic catalogs, but we have changed a few items which we will highlight. We also include some commentary on the parameter values in annotations within the example files.

To run SExtractor, the default.* files must be in the working directory, and set up with parameter values as desired. Then, bearing in mind the offset in extension number (assuming use of SExtractor version 2.8.6 as was used here), and specifying '' as the catalog name and the checkimage name in the default.sex file (as shown in Appendix B), the commands to issue are simply of the form below (the two-line commands wrap)

```
sex -catalog_name  jabu01fvq_crclean_ext1.cat  -checkimage_name
jabu01fvq_crclean_ext1_check.fits  jabu01fvq_crclean.fits'[0]'
sex  -catalog_name jabu01fvq_crclean_ext4.cat  -checkimage_name
jabu01fvq_crclean_ext4_check.fits  jabu01fvq_crclean.fits'[3]'
etc.
```

such that a separate catalog and checkimage is generated for each of the two chips for each image since the **TweakReg** task expects its input in that format. You could instead just issue the commands above in the format of "sex jabu01fvq_crclean.fits'[0]' " if you edited the name of the output catalog and checkimage as appropriate in the default.sex file before each issuance of the command to run SExtractor.

As determined by the **default.params** file, the specific format of the output catalog in this example is given at the head of the output catalog, and is the following, with the first number on the left representing the column number in the output:

```
#   1 X_IMAGE          Object position along x                    [pixel]
#   2 Y_IMAGE          Object position along y                    [pixel]
#   3 MAG_AUTO         Kron-like elliptical aperture magnitude    [mag]
#   4 CLASS_STAR       S/G classifier output
#   5 NUMBER           Running object number
#   6 FLAGS            Extraction flags
```

The most critical components of the output are x and y measurements of the objects, plus magnitude or flux information. Depending on whether the flux values in the images are expressed in terms of counts or counts per second, the magnitude zeropoint may be expressed in different ways. In the case of this example, the images were in counts, so the magnitude zeropoint was calculated as (M + 2.5 log (expoure time)) where M was a value obtained from http://www.stsci.edu/hst/acs/analysis/zeropoints/zpt.py which is the web page for the ACS zero point calculator. This will enable later optional automatic trimming (within **TweakReg**) of the input list for matching based on magnitude limits. In this case, the ST Mag system was used, but it could easily have been any of several other magnitude systems which are offered as options in the ACS zero point calculator. The important thing here is the relative difference in magnitudes between objects within any given system since it is used in **TweakReg** for setting faint and bright limits on the list of objects to include for matching.

If the information about FWHM and pixel scale was used such that SExtractor stellarity values are valid, then that quantity may also be used to eliminate some objects from the matching solution, if desired, by the use of awk commands, etc. Other options for trimming the object lists in the input catalogs include trimming boundaries of some number of pixels in x and y around the edges of the chips, also by the use of awk commands. An example of this will be shown in the section on Step 5, **Iterating in TweakReg**.

It is always a good idea to actually look at some catalogs overplotted on your images to see if they actually seem to be reasonable, picking up real objects in a reasonable magnitude range, getting reasonable-looking centers for them, and discriminating well between sources in complex areas with multiple objects, etc. To do this, with the use of awk commands, for example, on can easily turn a copy of input catalogs into files of ds9 region format and then overplot them on the single-chip images. Examples of two of our initial catalogs are shown in Figures 1 and 2, representing the 2 different chips of one of the *_crclean.fits images, while Figures 3 and 4 show some of the details.

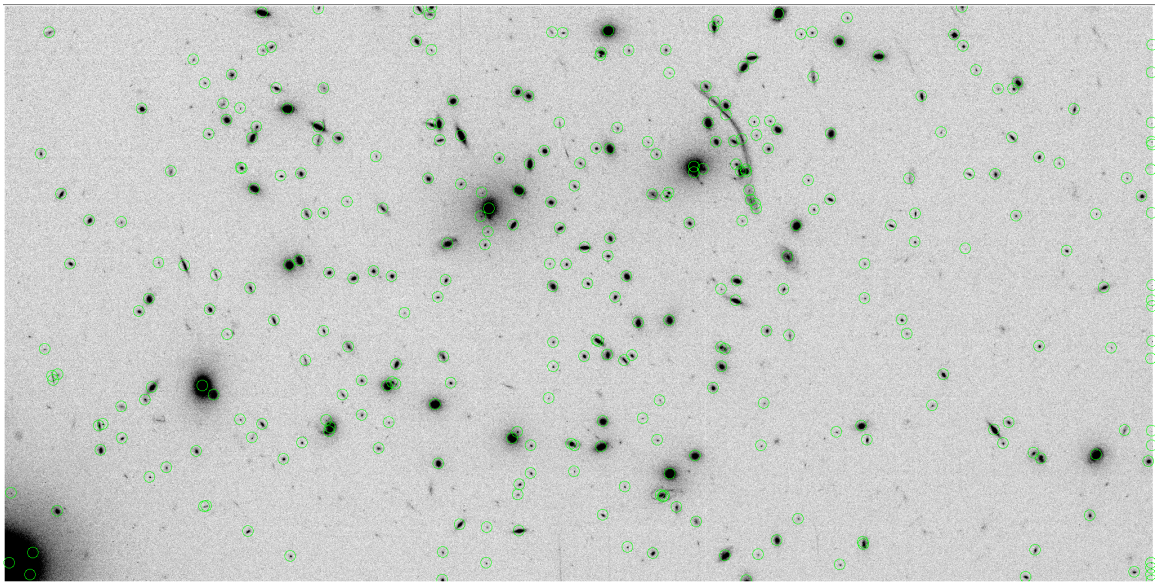

Figure 1: Extension 1 of one of the images, above, and the original SExtractor output catalog overplotted in green.

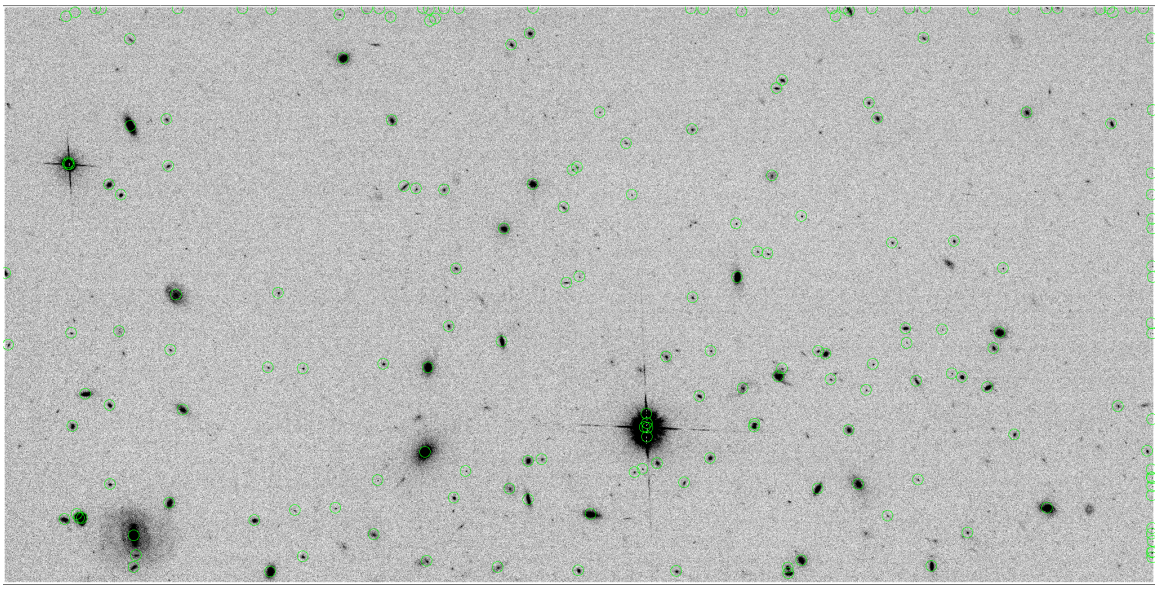

Figure 2: Extension 4 of one of the images, above, and the original SExtractor output catalog overplotted in green.

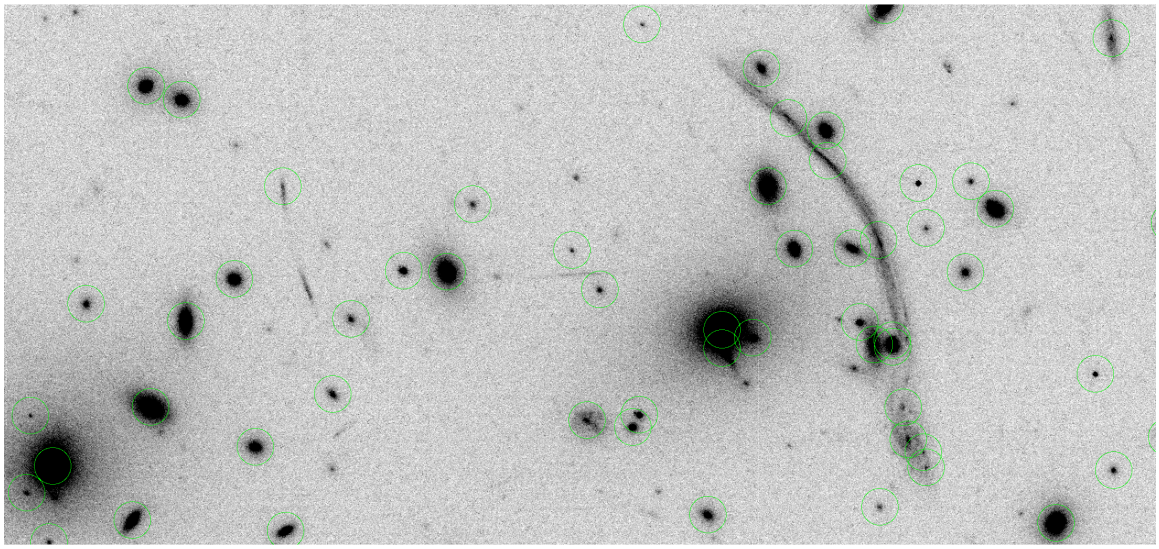

Figure 3: Close-up, above, showing detail of some of the SExtractor catalog objects, including some in the lensed arc.

Some spurious objects are visible in the lensed arc in Extension 1 and in the bright star in Extension 4. However, most catalog object entries look real and reasonable when looking at the objects in the images on which they are overplotted. Such sanity checks are always important to make, and can also help inform choices of objects to delete in an attempt to find a better solution when doing iterative **TweakReg** runs.

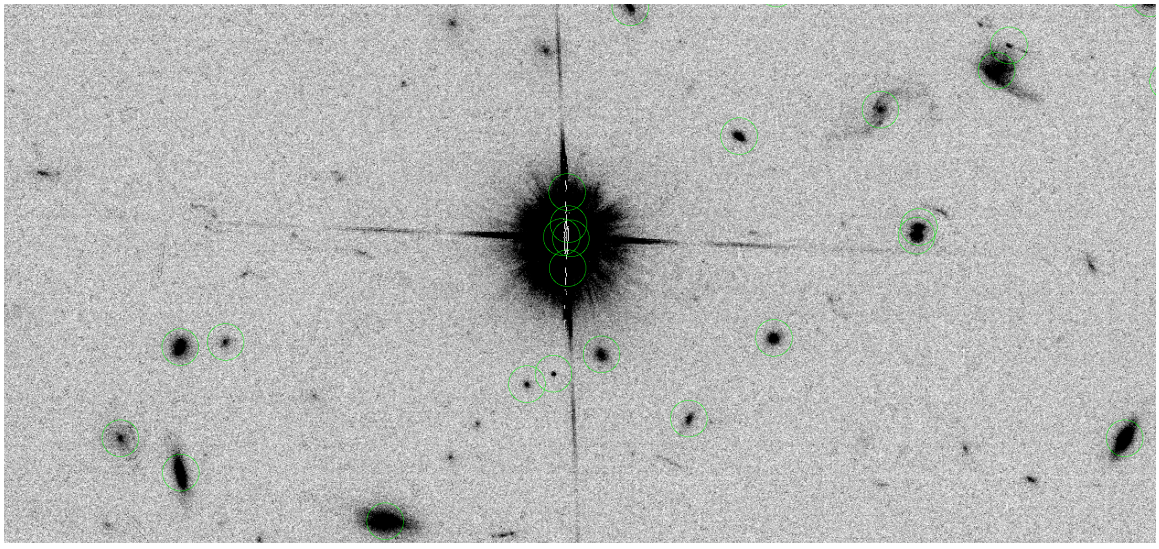

Figure 4: Close-up, above, showing detail of some of the SExtractor catalog objects, including some which are artifacts associated with the bright star.

Step 4: Feed SExtractor Catalogs to TweakReg

Once you have created the SExtractor catalogs on the basis above (a separate catalog for each chip of each image), you can then feed them to the **TweakReg** task in order for it to do its source matching and alignment. It is usually an iterative process, so you will usually run **TweakReg** multiple times with the *updatehdr* keyword set to "False" (No") during those iterations, and only set it to "True" ("Yes") in a final re-run of **TweakReg** when you have found a solution which you consider acceptable and have set the task parameters and version of the input catalogs to match the conditions which produced the most desired results. Remember also that you should not update the WCS again, or you will undo the geometric corrections you have applied earlier.

To begin the alignment process, **TweakReg** expects a file which lists the images to be aligned, the *_flc.fits images, along with the catalogs from each chip of the *_crclean.fits images in the form "flc.fits crclean-chip1.cat crclean-chip2.cat" thus:

```
jabu01fvq_flc.fits   jabu01fvq_crclean_ext1.cat   jabu01fvq_crclean_ext4.cat
jabu01gfq_flc.fits   jabu01gfq_crclean_ext1.cat   jabu01gfq_crclean_ext4.cat
jabu01gmq_flc.fits  jabu01gmq_crclean_ext1.cat  jabu01gmq_crclean_ext4.cat
jabu01gsq_flc.fits  jabu01gsq_crclean_ext1.cat   jabu01gsq_crclean_ext4.cat
jabu01gyq_flc.fits  jabu01gyq_crclean_ext1.cat  jabu01gyq_crclean_ext4.cat
jabu01hcq_flc.fits  jabu01hcq_crclean_ext1.cat  jabu01hcq_crclean_ext4.cat
```

This file may be called by any name, as long as that name is given as the parameter value to the "*catfile*" parameter in the **TweakReg** task. In this case, even though we are feeding it our own external SExtractor catalogs, we have chosen to name it just as the **TweakReg** task names it by default when it uses its own internally-generated catalogs from the **ImageFindPars** task, "coords_catfile.lis".

We begin the iterative process by running **TweakReg** with mostly default parameter values to first see what kind of result we get that way, and then iterate further if we feel it is needed, or just to see if we can easily improve on the initial solution. Our initial run task parameters for **TweakReg** are shown in Appendix F. Note that, though we don't use it directly, just for our own informational purposes, we produce a differently-named (by version number and other distinguishing characteristics, etc.) shift file and wcs.fits file for each tentative run even though we are not updating the header with any improved solutions at this stage. This allows us to get a better sense of how consistently different iterations are finding very nearly the same shifts, or where they may deviate more significantly, and to see the x and y rms values for each image in that run, to which we can refer, along with the plots for each run which we also save for comparison, naming

them by the most distinguishing part of their rootname and iteration or version number and any other particular differences we made in the input catalogs, etc.

An example of the shift file we obtain from our initial run is here, below. From left to right, the quantities reported are image, x-shift, y-shift, rotation, scale, x-rms, and y-rms (x and y shifts and rms units are in pixels, rotation is in degrees).

```
# frame: output
# refimage: shifts-ver1_wcs.fits[wcs]
# form: delta
# units: pixels
jabu01fvq_flc.fits   0.000000  0.000000     0.000000   1.000000  0.000000  0.000000
jabu01gfq_flc.fits   0.022882  -0.024646   359.999418   1.000002  0.160056  0.158230
jabu01gmq_flc.fits   0.082594  0.001245    359.998965   1.000009  0.165687  0.163800
jabu01gsq_flc.fits   0.081501  -0.004551   359.998707   1.000003  0.139776  0.153342
jabu01gyq_flc.fits   0.169839  0.060228    359.998981   1.000007  0.146863  0.156113
jabu01hcq_flc.fits   0.133796  0.109905    359.999174   1.000007  0.147139  0.157516
```

We also obtain plots for each of the images being compared to its reference, which by default is the first image in the series. Examples of the types of plots we obtain for one of the images in the series are shown below in Figures 5, 6, and 7.

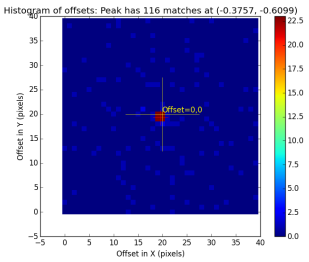

Figure 5: Typical initial histogram of offsets.

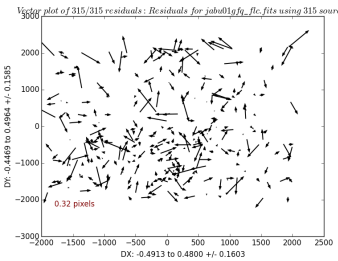
Vector plot of 315/315 residuals: Residuals for jabu01gfq_flc.fits using 315 sources
0.32 pixels
DX: -0.4913 to 0.4800 +/- 0.1603
DY: -0.4469 to 0.4964 +/- 0.1585

Figure 6: Typical initial vector plot of residuals.

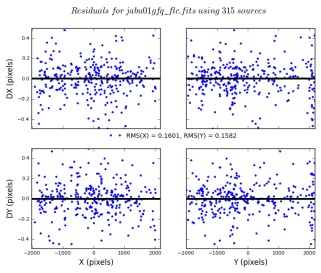

Figure 7: Typical initial x,y rms (pixel) residuals

Step 5: Iterating in TweakReg

As mentioned above, one may want to see if any better solutions can be easily obtained, and that some cuts to the input object lists in the catalogs may be made using the built-in flux or magnitude limit parameters within the TweakReg task itself. And, as also mentioned above, an awk command to trim catalogs and select objects based on greater or lesser stellarity may also be used. Early-type galaxies (ellipticals, etc.) may be better sources to use for alignment because they are less prone to changes in

morphological appearance over the wavelength range of images in several filters, but stars may also be bad sources to use for alignment if saturated, or if spurious sources were found around them in their diffraction spikes or airy rings or other artifacts in the PSF, or if they have a high enough proper motion and if images were taken far enough apart in time that the stars have moved between the frames. (This is not the case here in this example, and it is generally only the case when observations are separated by some number of years for the highest proper motion stars in most fields.) Appendix G represents an example of a **TweakReg** parameter file from one of the iterations. If we manually deleted any objects from the object list, it will not contain parameters for that.

One very important thing to mention regarding automatic trimming of the input catalogs within **TweakReg** based on flux limits or magnitude limits is the correct usage of the *maxflux* and *minflux* parameter values in **TweakReg**. If the catalog values are simply in flux units, then *maxflux* and *minflux* may be used simply and intuitively just as they would seem to imply. In the case of using regular flux measurements, a larger number corresponds to a brighter source and a smaller number corresponds to a fainter source. However, if the SExtractor catalog flux values have been converted into magnitudes as is the case in this example, then the use of *maxflux* and *minflux* can be a bit counterintuitive. It must be remembered that brighter magnitudes correspond to smaller numbers and fainter magnitudes correspond to larger numbers, so that, when using magnitudes as input in the catalog files, the software is written and works such that you must use what seems to be a reverse order for maximum and minimum, corresponding to the size of the number rather than the magnitude of the source. For example, when using magnitudes rather than regular flux measurements, you would specify *maxflux* = 24.0 and *minflux* = 20.0 to select and trim the magnitude range to include only sources from magnitude 20.0 to 24.0 in your input SExtractor catalog for use in matching sources although the name of the parameters make that seem counterintuitive and to prefer the opposite. One will encounter crashes with errors if using the values for these parameters in the same sense for magnitudes as for regular fluxes.

Another kind of object filtering one may want to do is to trim the objects around the edge of the chips if they happen to be very close to the edge or suspect in those areas for some other reason. This may be useful if there are areas in the *_crclean.fits images which did not have overlap with other images and there are areas around edges which were not effectively cleaned of cosmic rays and those have dominated the object lists there, or if there are other spurious or questionable objects detected along the edges of an image. For example, given the specific format of our output SExtractor catalogs which was set by what we have defined in our **default.params** file, the awk syntax for trimming objects in a 100-pixel border around the edges of the ACS/WFC chips which are 4096 x 2048 pixels each would be this. (NOTE: The line below wraps.)

```
awk '($1 !~ /#/ && $1>100 && $1<3996 && $2>100 && $2<1948)'
jabu01fvq_crclean_ext1.cat > jabu01fvq_crclean_trimv1_ext1.cat
```

The $1 represents the column of x values and $2 the column of y values. From this example, one may also anticipate part of the command necessary to trim the list based on stellarity, etc. except for which stellarity values to use as a cut-off.

Whether we trimmed our input catalog object lists by magnitude ranges or by trimming some portion of a border area around the chip edges or by stellarity or manually by some other means and based on some other criteria, to use the trimmed object catalogs in **TweakReg**, the resulting new catalogs must then be referenced in the **TweakReg** catfile parameter in a new file like the coords_catfile.lis which was used for the original **TweakReg** run. We could contineu to call it coords_catfile.lis or any other name we wish, but to most accurately describe it, we'll call it coordstrimmed_catfile.lis, and it has the same format as the original coords_catfile.lis, but references the new, trimmed catalogs instead, as below. (NOTE: The lines below wrap, making a total of 6 lines.)

```
jabu01fvq_flc.fits jabu01fvq_crclean_trimv1_ext1.cat
jabu01fvq_crclean_trimv1_ext4.cat
jabu01gfq_flc.fits jabu01gfq_crclean_trimv1_ext1.cat
jabu01gfq_crclean_trimv1_ext4.cat
jabu01gmq_flc.fits jabu01gmq_crclean_trimv1_ext1.cat
jabu01gmq_crclean_trimv1_ext4.cat
jabu01gsq_flc.fits jabu01gsq_crclean_trimv1_ext1.cat
jabu01gsq_crclean_trimv1_ext4.cat
jabu01gyq_flc.fits jabu01gyq_crclean_trimv1_ext1.cat
jabu01gyq_crclean_trimv1_ext4.cat
jabu01hcq_flc.fits jabu01hcq_crclean_trimv1_ext1.cat
jabu01hcq_crclean_trimv1_ext4.cat
```

It should also be remembered that, although automation of object inclusion or exclusion in the source lists is generally most desirable, if need be, the input object lists can even be manually edited to include or exclude any particularly useful or troublesome objects, as the case may be. Quoting from Paper I (ACS ISR 2015-04): "You may also want to try deleting from your catalogs troublesome objects which have high residuals as shown in the **TweakReg** plots, but that, too, can be tricky, and the size of an object's residuals may vary some from image to image and catalog to catalog. In general, it is perhaps better to use other methods, such as scripted awk commands to remove such objects on the periphery or elsewhere, or within specified magnitude ranges or stellarity

values, or ellipticity, or galaxy morphology type, etc. (Small elliptical galaxies may be better than galaxies with more complex morphology when having to compare across different bandpasses, for example.)" However, simple removal of objects with high residuals may also work very well since they are, after all, seemingly problematic in at least some of the images and catalogs.

In fact, although a tedious process which most would not want to try in most circumstances, in the case of badly mis-aligned images with grossly bad pointing info in the header, etc. due to something like a guide star problem during acquisition, manual measurement and generation of catalog entries for objects in such an image which are then inserted into the input catalog(s) can be an effective way of correcting the header info in those images and bringing them into a close enough spatial concordance with the others that the **TweakReg** task can then be used in subsequent iterations to more finely align them to other images at similar pointings. In fact, in even more labor-intensive fashion, an entire manually-measured and manually-edited list of catalog object positions could be created for each image and run through **TweakReg** in this manner if it was advantageous, but that is, as said, a very labor-intensive exercise which is usually much better automated in some way if possible, and so we only mention the possibility of it here and leave the rest to the reader as an exercise if so desired.

Step 6: Re-running TweakReg to Update the Headers of Files to Be Aligned

Once a solution has been found that the user deems acceptable, or at least the best possible to achieve with the current process, if it is desired to continue with that alignment, the **TweakReg** task must be run once more with the *updatehdr* keyword set to "Yes" in order to apply the corrections to all the *_flc.fits images in that same filter/same visit set. Appendix H shows a version of such ultimate **TweakReg** parameters, with the main difference being that one keyword value for *updatehdr* being set to "Yes" ("True").

Step 7: The Remainder of the Process

We will not cover all of the rest of the process in detail here, but will merely briefly describe it since it is re-integrated with the rest of the process that would normally be used had the input catalogs been those generated internally in **TweakReg** via the use of the built-in **ImageFindPars** task.

Once you have internally aligned each same visit/same filter combination of images, you can then drizzle a set of them in one filter to a "final" *_drc.fits image which you can then use as a reference image to transfer the solution to all the _flc.fits images across all filters and all visits via the **DrizzlePac TweakBack** task. Those images can then all be

drizzled together on a per-filter basis, with the resulting images all being aligned together. This process is described elsewhere in other examples and documentation and will not be described further here. The reader is referred to that documentation instead.

The Best SExtractor Options

Although varying SExtractor object detection and measuring parameters and also trimming catalog lists by magnitude limits, spatial chip-edge limits, and stellarity, as well as by manual massaging of individual case-by-case circumstances of various sorts which may help to make a final solution somewhat better, it is generally the case that the most effective way to make the coordinate measures of objects the most reliable and consistent in SExtractor and improve the solutions by a significant amount is to incorporate the use of the relevant weight map (always in IVM format, as required by SExtractor) when detecting and measuring the positions of objects. The use of a weight map can easily be specified on the command line when running SExtractor. It can potentially be very helpful in almost all circumstances, but perhaps especially so in cases such as when measuring objects taken across different visits and in mosaics or stacks where there are areas of varying depth and signal-to-noise. This practice will be addressed in more detail elsewhere, including in the Frontier Fields technical documentation and scripts when published.

Using inverse variance (IVM) weight maps when available is probably the single most beneficial improvement one can make in obtaining better coordinates and registration. Along with that, the kinds of things described in this ISR including trimming the list of objects by removing fainter objects, extraordinarily bright objects such as bright stars, objects on the edges of the image which may not have been measured as well, objects with complex morphologies which may also vary more greatly from bandpass to bandpass (although they may be OK internally within a single filter), and, in the case of images taken over a long time baseline, removal of stars which may have proper motion - all of these and still more things represent areas which may be investigated/tried for possible improvement of the solution. All datasets and combinations of datasets are different and each present their own unique challenges.

Conclusions

We have shown examples of parameter files for **DrizzlePac's AstroDrizzle** and **TweakReg** tasks as they may be used in preparing cosmic ray-cleaned images for the use of SExtractor on them, and in incorporating SExtractor-produced object catalogs for use in image alignment and registration. We have also shown examples of the four basic SExtractor startup files, **default.sex**, **default.conv**, **defaul.nnw**, and **default.params** as

they might be used on datasets of this sort. We have also shown examples of other files such as the SExtractor catalog format and the image plus catalog file lists such as coords_catfile.lis needed in the process. Finally, as an aid to using these tools, we have also given advice on some general guidelines for items such as suggested values for some critical parameters, and we have given advice on some things to explore in iterative efforts to improve astrometric solutions beyond their initial values, to see if any improvement is readily possible.

Recommendations

In situations such as we have described where SExtractor is needed, we recommend the creation of cosmic ray-cleaned images as input for SExtractor. We recommend the use of SExtractor version 2.8.6 or later. In SExtractor's **default.sex** file, the *back_size* should be a little larger than an average, typical small, faint galaxy in number of pixels. A value of 64 is a good value for small HUDF-like galaxies, together with a *back_filtersize* of 3 or so. A *detect_minarea* of 3 and *detect_thresh* of 7 are similarly reasonable values, and *analysis_thresh* can sometimes also be useful for experimentation. Cuts in the lists based on magnitude, (x,y) position, stellarity, and several other edits may help make the solution better, but ultimately, the use of inverse variance weight maps is one of the best aids to better solutions.

Acknowledgements

We are grateful to Mihai Cara, Warren Hack, and fellow members of the STScI DrizzlePac/AstroDrizzle Working Group for advice and a careful reading of this paper.

References

SExtractor:

SExtractor: http://www.astromatic.net/software/sextractor
Bertin, E., & Arnouts, S., 1996: SExtractor: Software for Source Extraction, Astronomy & Astrophysics Supplement 317, 393
Bertin, E., SExtractor v2.13 User's Manual, at
https://www.astromatic.net/pubsvn/software/sextractor/trunk/doc/sextractor.pdf
Akhlaghi, M., 2012, Notes on SExtractor 2.8.6, at
http://www.astr.tohoku.ac.jp/~akhlaghi/sextractor_notes.html
Holwerda, B., SExtractor for Dummies, at
http://astroa.physics.metu.edu.tr/MANUALS/sextractor/Guide2source_extractor.pdf

DrizzlePac tasks and usage:
DrizzlePac tasks and usage:
Gonzaga et al., 2012, The DrizzlePac Handbook (Available on-line.)
http://documents.stsci.edu/hst/HST_overview/documents/DrizzlePac/DrizzlePac.cover.html

STScI DrizzlePac Web Site

http://www.stsci.edu/hst/HST_overview/drizzlepac

AAS Posters at DrizzlePac web site:

Lucas, R.A., 2012, AstroDrizzle: Utilizing New CALACS Products, AAS-Anchorage, Alaska http://www.stsci.edu/hst/HST_overview/drizzlepac/resources/ral_aas-anchorage_2012_template-v7_Lucas.pdf

Lucas, R.A., Hilbert, B., and Hack, W. 2013, Using SExtractor With the DrizzlePac Tweakreg Software for Aligning and Combining Sources in HST/ACS Images, AAS-Long Beach, California
http://www.stsci.edu/hst/HST_overview/drizzlepac/resources/Lucas_AAS.pdf

ISRs

Lucas, R.A. and Hilbert, B., 2015, Basic Use of SExtractor Catalogs With TweakReg – I, Instrument Science Report ACS 2015-04

Appendix A – Initial AstroDrizzle Input Parameters:

```
_task_name_ = astrodrizzle#
input = *flc.fits# Input files (name, suffix, or @list)
output = ""# Rootname for output drizzled products
runfile = astrodrizzle.log# File for logging the processing
updatewcs = False# Update the WCS keywords?
wcskey = ""# WCS version to use in processing
proc_unit = native# Units used during processing
coeffs = True# Use header-based distortion coefficients?
context = True# Create context image during final drizzle?
group = ""# Single extension or group to be combined/cleaned
build = False# Create multi-extension output file for final drizzle?
crbit = 4096# Bit value for CR ident. in DQ array
stepsize = 10# Step size for drizzle coordinate computation
resetbits = 4096# Bit values to reset in all input DQ arrays
num_cores = None# Max CPU cores to use (n<2 disables, None = auto-decide)
in_memory = False# Process everything in memory to minimize disk I/O?

[STATE OF INPUT FILES]
restore = False# Copy input files FROM archive directory for processing?
preserve = True# Copy input files to archive directory, if not already archived?
overwrite = False# Copy input files into archive, overwriting if required?
clean = False# Delete temporary files after completion?

[STEP 1: STATIC MASK]
static = True# Create static bad-pixel mask from the data?
static_sig = 4.0# "Sigma*rms below mode to clip for static mask"

[STEP 2: SKY SUBTRACTION]
skysub = True# "Perform sky subtraction?"
skywidth = 0.1# "Bin width for sampling sky statistics (in sigma)"
skystat = median# "Sky correction statistics parameter"
skylower = None# "Lower limit of usable data for sky (always in electrons)"
skyupper = None# "Upper limit of usable data for sky (always in electrons)"
skyclip = 5# "Number of clipping iterations"
skylsigma = 4.0# Lower side clipping factor (in sigma)
skyusigma = 4.0# Upper side clipping factor (in sigma)
skyuser = ""# KEYWORD indicating a sky subtraction value if done by user.
skyfile = ""# Name of file with user-computed sky values.
skymethod = localmin# Sky computation method
skymask_cat = ""# Catalog file listing image masks
use_static = True# "Use static mask for skymatch computations?"
sky_bits = 0# Integer mask bit values considered good pixels in DQ array

[STEP 3: DRIZZLE SEPARATE IMAGES]
driz_separate = True# "Drizzle onto separate output images?"
driz_sep_kernel = turbo# Shape of kernel function
driz_sep_wt_scl = exptime# "Weighting factor for input data image"
driz_sep_pixfrac = 1.0# Linear size of drop in input pixels
driz_sep_fillval = None# Value to be assigned to undefined output points
driz_sep_bits = 0# Integer mask bit values considered good
driz_sep_compress = False# "Use compression when writing out product?"

[STEP 3a: CUSTOM WCS FOR SEPARATE OUTPUTS]
driz_sep_wcs = False# "Define custom WCS for separate output images?"
driz_sep_refimage = ""# Reference image from which to obtain a WCS
driz_sep_rot = None# "Position Angle of drizzled image's Y-axis w.r.t. North (degrees)"
driz_sep_scale = None# Absolute size of output pixels in arcsec/pixel
driz_sep_outnx = None# Size of separate output frame's X-axis (pixels)
driz_sep_outny = None# Size of separate output frame's Y-axis (pixels)
driz_sep_ra = None# right ascension output frame center in decimal degrees
driz_sep_dec = None# declination output frame center in decimal degrees

[STEP 4: CREATE MEDIAN IMAGE]
median = True# "Create a median image?"
median_newmasks = True# "Create new masks when doing the median?"
combine_maskpt = 0.3# "Percentage of weight image value below which it is flagged as a bad pixel."
combine_type = iminmed# "Type of combine operation"
combine_nsigma = 4 3# "Significance for accepting minimum instead of median"
combine_nlow = 0# "minmax: Number of low pixels to reject"
combine_nhigh = 0# "minmax: Number of high pixels to reject"
combine_lthresh = None# Lower threshold for clipping input pixel values
combine_hthresh = None# "Upper threshold for clipping input pixel values"
combine_grow = 1# Radius (pixels) for neighbor rejection
combine_bufsize = None# "Size of buffer(in Mb) for each input image"
```

(Initial AstroDrizzle Input Parmeters – continued)

```
[STEP 5: BLOT BACK THE MEDIAN IMAGE]
blot = True# "Blot the median back to the input frame?"
blot_interp = poly5# Interpolant (nearest,linear,poly3,poly5,sinc)
blot_sinscl = 1.0# Scale for sinc interpolation kernel
blot_addsky = True# "Add sky using MDRIZSKY value from header?"
blot_skyval = 0.0# Custom sky value to be added to blot image

["STEP 6: REMOVE COSMIC RAYS WITH DERIV, DRIZ_CR"]
driz_cr = True# Perform CR rejection with deriv and driz_cr?
driz_cr_corr = True# "Create CR cleaned _crclean file and a _crmask file?"
driz_cr_snr = 3.5 3.0# "Driz_cr.SNR parameter"
driz_cr_grow = 1# Driz_cr_grow parameter
driz_cr_ctegrow = 0# Driz_cr_ctegrow parameter
driz_cr_scale = 1.2 0.7# Driz_cr.scale parameter

[STEP 7: DRIZZLE FINAL COMBINED IMAGE]
driz_combine = True# "Perform final drizzle image combination?"
final_wht_type = IVM# Type of weighting for final drizzle
final_kernel = square# Shape of kernel function
final_wt_scl = exptime# Weighting factor for input data image
final_pixfrac = 1.0# Linear size of drop in input pixels
final_fillval = None# "Value to be assigned to undefined output points"
final_bits = 0# Integer mask bit values considered good
final_units = cps# Units for final drizzle image (counts or cps)

[STEP 7a: CUSTOM WCS FOR FINAL OUTPUT]
final_wcs = False# "Define custom WCS for final output image?"
final_refimage = ""# Reference image from which to obtain a WCS
final_rot = None# "Position Angle of drizzled image's Y-axis w.r.t. North (degrees)"
final_scale = None# Absolute size of output pixels in arcsec/pixel
final_outnx = None# "Size of FINAL output frame X-axis (pixels)"
final_outny = None# Size of FINAL output frame Y-axis (pixels)
final_ra = None# right ascension output frame center in decimal degrees
final_dec = None# declination output frame center in decimal degrees

[INSTRUMENT PARAMETERS]
gain = ""#
gnkeyword = ""#
rdnoise = ""#
rnkeyword = ""#
exptime = ""#
expkeyword = ""#
```

NOTE: We have turned on Step 7, the final drizzle, here, for our own curiosity, even before the fine alignment with **TweakReg**, just to see how a rough first version of a final image might look in this case where all of the images in all of the filters were taken in the same visit, but this need not be done for the process we have described here.

Appendix B – Basic SExtractor default.sex file Parameters:

```
# Default configuration file for SExtractor 2.8.6
# EB 2010-11-16
#
# Modified to produce output files for use as input files (after further
# editing) to the Drizzlepac tweakreg task.
#
# Edit the CATALOG_NAME below to use the distinctive part of the file name
# for each unique image in the group of images which you are cataloging.
# Do this for each new catalog run, once for each image, at least as the
# SExtractor software seems to run on MEF files for now.
#
# Edit the CHECKIMAGE_NAME below to use the distinctive part of the file name
# for each unique image in the group of images which you are cataloging.
# Do this for each new catalog run, once for each image, at least as the
# SExtractor software seems to run on MEF files for now.
#
# The CATALOG_NAME and CHECKIMAGE_NAME should be in sync and have the same
# rootname for each of your runs so that you have a catalog and checkimage
# with the same rootname as each other for each of your successive runs.
#
# Edit the default.param file to make any changes you desire (or that tweakreg
# requires) for the output catalog which you will edit and then feed to the
# tweakreg task. You will typically use the catalogs from the first file as
# the reference image for all the rest, and may copy it to a name reflective
# of that, as such, or may leave it with its regular name, but just specify it
# as the reference image in Drizzlepac tweakreg when you run that.
#
# Ray A. Lucas - 31 May, 2012
#
#
#-------------------------------- Catalog ------------------------------------

# Give this in the command line instead.
#CATALOG_NAME     jabu01fvq_crclean_ext1.cat       # name of the output catalog
CATALOG_NAME     ''            # name of the output catalog

CATALOG_TYPE     ASCII_HEAD     # NONE,ASCII,ASCII_HEAD, ASCII_SKYCAT,
                                # ASCII_VOTABLE, FITS_1.0 or FITS_LDAC
PARAMETERS_NAME  default.param  # name of the file containing catalog contents

#------------------------------- Extraction ----------------------------------

DETECT_TYPE      CCD            # CCD (linear) or PHOTO (with gamma correction)
#DETECT_MINAREA   5              # minimum number of pixels above threshold
DETECT_MINAREA   3              # minimum number of pixels above threshold
#DETECT_THRESH    1.5           # <sigmas> or <threshold>,<ZP> in mag.arcsec-2
DETECT_THRESH    7.0            # <sigmas> or <threshold>,<ZP> in mag.arcsec-2
ANALYSIS_THRESH  1.5            # <sigmas> or <threshold>,<ZP> in mag.arcsec-2

FILTER           Y              # apply filter for detection (Y or N)?
FILTER_NAME      default.conv   # name of the file containing the filter

DEBLEND_NTHRESH  32             # Number of deblending sub-thresholds
DEBLEND_MINCONT  0.005          # Minimum contrast parameter for deblending

CLEAN            Y              # Clean spurious detections? (Y or N)?
CLEAN_PARAM      1.0            # Cleaning efficiency

MASK_TYPE        CORRECT        # type of detection MASKing: can be one of
                                # NONE, BLANK or CORRECT
```

(Continued on next page)

(SExtractor defaul.sex file – continued)

```
#------------------------------- Photometry ---------------------------------

PHOT_APERTURES   5              # MAG_APER aperture diameter(s) in pixels
PHOT_AUTOPARAMS  2.5, 3.5       # MAG_AUTO parameters: <Kron_fact>,<min_radius>
PHOT_PETROPARAMS 2.0, 3.5       # MAG_PETRO parameters: <Petrosian_fact>,
                                # <min_radius>
# This is the average ACS/WFC saturation level of 84,700 in electrons divided by GAIN (2.0) for WFC1 and WFC2.
SATUR_LEVEL      42350.0        # level (in ADUs) at which arises saturation
SATUR_KEY        SATURATE       # keyword for saturation level (in ADUs)
# ST Mag Zeropoint taken from ACS ZP calculator for counts image (not cps in this case), so (calculator STmag value + 2.5logv10(exposure time), so (25.746 + 2.5log(1130.0))
MAG_ZEROPOINT    33.3787        # magnitude zero-point
MAG_GAMMA        4.0            # gamma of emulsion (for photographic scans)
# Post-SM4 ACS/WFC Default Gain is 2.0. For Pre-SM4 ACS/WFC data the default was Gain = 1.
GAIN             2.0            # detector gain in e-/ADU
GAIN_KEY         GAIN           # keyword for detector gain in e-/ADU
# Native ACS/WFC scale = 0.05 arcsec per pixel or you can leave as 0 to automatically use the FITS WCS info as below. Only used for CLASS_STAR calculation.
PIXEL_SCALE      0.05           # size of pixel in arcsec (0=use FITS WCS info)

#------------------------ Star/Galaxy Separation ----------------------------

#SEEING_FWHM      1.2            # stellar FWHM in arcsec
# FWHM for stellar source in ACS/WFC image should be ~1 pixel, and default drizzle pixel scale for ACS/WFC is 0.05arcsec/pixel. Can allow a bit so try 0.08.
SEEING_FWHM      0.08           # stellar FWHM in arcsec. Only used for CLASS_STAR calculation.
STARNNW_NAME     default.nnw    # Neural-Network_Weight table filename

#------------------------------- Background ---------------------------------

# Back_size should be a little larger than your average, typical small, faint galaxy in number of pixels.
BACK_SIZE        64             # Background mesh: <size> or <width>,<height>
BACK_FILTERSIZE  3              # Background filter: <size> or <width>,<height>

BACKPHOTO_TYPE   GLOBAL         # can be GLOBAL or LOCAL

#------------------------------- Check Image --------------------------------

CHECKIMAGE_TYPE  OBJECTS           # can be NONE, BACKGROUND, BACKGROUND_RMS,
                                # MINIBACKGROUND, MINIBACK_RMS, -BACKGROUND,
                                # FILTERED, OBJECTS, -OBJECTS, SEGMENTATION,
                                # or APERTURES
# Give this in the command line instead.
#CHECKIMAGE_NAME  jabu01fvq_crclean_ext1_check.fits     # Filename for the check-image
CHECKIMAGE_NAME      ''           # Filename for the check-image

#--------------------- Memory (change with caution!) -------------------------

MEMORY_OBJSTACK  3000           # number of objects in stack
MEMORY_PIXSTACK  300000         # number of pixels in stack
MEMORY_BUFSIZE   1024           # number of lines in buffer

#----------------------------- Miscellaneous ----------------------------------

VERBOSE_TYPE     NORMAL         # can be QUIET, NORMAL or FULL
WRITE_XML        N              # Write XML file (Y/N)?
XML_NAME         sex.xml        # Filename for XML output
NTHREADS          0              # Number of simultaneous threads for
```

Appendix C – Basic SExtractor default.conv file

```
CONV NORM
# 3x3 ``all-ground'' convolution mask with FWHM = 2 pixels.
1 2 1
2 4 2
1 2 1
```

Appendix D – Basic SExtractor default.nnw file

```
NNW
# Neural Network Weights for the SExtractor star/galaxy classifier (V1.3)
# inputs:       9 for profile parameters + 1 for seeing.
# outputs:      ``Stellarity index'' (0.0 to 1.0)
# Seeing FWHM range: from 0.025 to 5.5'' (images must have 1.5 < FWHM < 5 pixels)
# Optimized for Moffat profiles with 2<= beta <= 4.

 3 10 10  1

-1.56604e+00 -2.48265e+00 -1.44564e+00 -1.24675e+00 -9.44913e-01 -5.22453e-01  4.61342e-02  8.31957e-01  2.15505e+00  2.64769e-01
 3.03477e+00  2.69561e+00  3.16188e+00  3.34497e+00  3.51885e+00  3.65570e+00  3.74856e+00  3.84541e+00  4.22811e+00  3.27734e+00

-3.22480e-01 -2.12804e+00  6.50750e-01 -1.11242e+00 -1.40683e+00 -1.55944e+00 -1.84558e+00 -1.18946e-01  5.52395e-01 -4.36564e-01 -5.30052e+00
 4.62594e-01 -3.29127e+00  1.10950e+00 -6.01857e-01  1.29492e-01  1.42290e+00  2.90741e+00  2.44058e+00 -9.19118e-01  8.42851e-01 -4.69824e+00
-2.57424e+00  8.96469e-01  8.34775e-01  2.18845e+00  2.46526e+00  8.60878e-02 -6.88080e-01 -1.33623e-02  9.30403e-02  1.64942e+00 -1.01231e+00
 4.81041e+00  1.53747e+00 -1.12216e+00 -3.16008e+00 -1.67404e+00 -1.75767e+00 -1.29310e+00  5.59549e-01  8.08468e-01 -1.01592e-02 -7.54052e+00
 1.01933e+01 -2.09484e+01 -1.07426e+00  9.87912e-01  6.05210e-01 -6.04535e-02 -5.87826e-01 -7.94117e-01 -4.89190e-01 -8.12710e-02 -2.07067e+01
-5.31793e+00  7.94240e+00 -4.64165e+00 -4.37436e+00 -1.55417e+00  7.54368e-01  1.09608e+00  1.45967e+00  1.62946e+00 -1.01301e+00  1.13514e-01
 2.20336e-01  1.70056e+00 -5.20105e-01 -4.28330e-01  1.57258e-03 -3.36502e-01 -8.18568e-02 -7.16163e+00  8.23195e+00 -1.71561e-02 -1.13749e+01
 3.75075e+00  7.25399e+00 -1.75325e+00 -2.68814e+00 -3.71128e+00 -4.62933e+00 -2.13747e+00 -1.89186e-01  1.29122e+00 -7.49380e-01  6.71712e-01
-8.41923e-01  4.64997e+00  5.65808e-01 -3.08277e-01 -1.01687e+00  1.73127e-01 -8.92130e-01  1.89044e+00 -2.75543e-01 -7.72828e-01  5.36745e-01
-3.65598e+00  7.56997e+00 -3.76373e+00 -1.74542e+00 -1.37540e-01 -5.55400e-01 -1.59195e-01  1.27910e-01  1.91906e+00  1.42119e+00 -4.35502e+00

-1.70059e+00 -3.65695e+00  1.22367e+00 -5.74367e-01 -3.29571e+00  2.46316e+00  5.22353e+00  2.42038e+00  1.22919e+00 -9.22250e-01 -2.32028e+00


 0.00000e+00
 1.00000e+00
```

Appendix E – Basic SExtractor default.params file

```
# SExtractor default.param for output catalog file(s) to become
# Drizzlepac tweakreg input files.
# Ray A. Lucas - 31 May, 2012

# Re-ordering from default. Instead of Number, I start with these
# output parameters to order them in the format in which the new
# Drizzlepac tweakreg task requires them.
# Ray A. Lucas - 2012-05-31

X_IMAGE
Y_IMAGE
MAG_AUTO
CLASS_STAR
NUMBER

# Original below here, with duplicate (moved) items commented out.

#EXT_NUMBER - in SExtractor v2.8.6, all output extension numbers are "1", so useless to specify.
#NUMBER

#FLUX_ISO
#FLUXERR_ISO
#MAG_ISO
#MAGERR_ISO

#FLUX_ISOCOR
#FLUXERR_ISOCOR
#MAG_ISOCOR
#MAGERR_ISOCOR

#FLUX_APER(1)
#FLUXERR_APER(1)
#MAG_APER(1)
#MAGERR_APER(1)

#FLUX_AUTO
#FLUXERR_AUTO
#MAG_AUTO
#MAGERR_AUTO

#FLUX_BEST
#FLUXERR_BEST
#MAG_BEST
#MAGERR_BEST

#KRON_RADIUS
#BACKGROUND

#THRESHOLD
#MU_THRESHOLD
#FLUX_MAX
#MU_MAX
#ISOAREA_IMAGE
#ISOAREA_WORLD

#XMIN_IMAGE
#YMIN_IMAGE
#XMAX_IMAGE
#YMAX_IMAGE

#X_IMAGE
#Y_IMAGE
#X_WORLD
#Y_WORLD
#ALPHA_SKY
#DELTA_SKY
#ALPHA_J2000
#DELTA_J2000
#ALPHA_B1950
#DELTA_B1950

#X2_IMAGE
#Y2_IMAGE
#XY_IMAGE
#X2_WORLD
#Y2_WORLD
#XY_WORLD
```

(SExtractor default.params file – continued)

```
#------------------------------- Photometry ---------------------------------

PHOT_APERTURES   5              # MAG_APER aperture diameter(s) in pixels
PHOT_AUTOPARAMS  2.5, 3.5       # MAG_AUTO parameters: <Kron_fact>,<min_radius>
PHOT_PETROPARAMS 2.0, 3.5       # MAG_PETRO parameters: <Petrosian_fact>,
                                # <min_radius>
# This is the average ACS/WFC saturation level of 84,700 in electrons divided by GAIN (2.0) for WFC1 and WFC2.
SATUR_LEVEL      42350.0        # level (in ADUs) at which arises saturation
SATUR_KEY        SATURATE       # keyword for saturation level (in ADUs)
# ST Mag Zeropoint taken from ACS ZP calculator for counts image (not cps in this case), so (calculator STmag value + 2.5logv10(exposure time), so (25.746 + 2.5log(1130.0)
MAG_ZEROPOINT    33.3787        # magnitude zero-point
MAG_GAMMA        4.0            # gamma of emulsion (for photographic scans)
# Post-SM4 ACS/WFC Default Gain is 2.0. For Pre-SM4 ACS/WFC data the default was Gain = 1.
GAIN             2.0            # detector gain in e-/ADU
GAIN_KEY         GAIN           # keyword for detector gain in e-/ADU
# Native ACS/WFC scale = 0.05 arcsec per pixel or you can leave as 0 to automatically use the FITS WCS info as below. Only used for CLASS_STAR calculation.
PIXEL_SCALE      0.05           # size of pixel in arcsec (0=use FITS WCS info)

#------------------------ Star/Galaxy Separation ----------------------------

#SEEING_FWHM      1.2            # stellar FWHM in arcsec
# FWHM for stellar source in ACS/WFC image should be ~1 pixel, and default drizzle pixel scale for ACS/WFC is 0.05arcsec/pixel. Can allow a bit so try 0.08.
SEEING_FWHM      0.08           # stellar FWHM in arcsec. Only used for CLASS_STAR calculation.
STARNNW_NAME     default.nnw    # Neural-Network_Weight table filename

#------------------------------- Background ---------------------------------

# Back_size should be a little larger than your average, typical small, faint galaxy in number of pixels.
BACK_SIZE        64             # Background mesh: <size> or <width>,<height>
BACK_FILTERSIZE  3              # Background filter: <size> or <width>,<height>

BACKPHOTO_TYPE   GLOBAL         # can be GLOBAL or LOCAL

#------------------------------- Check Image --------------------------------

CHECKIMAGE_TYPE  OBJECTS           # can be NONE, BACKGROUND, BACKGROUND_RMS,
                                # MINIBACKGROUND, MINIBACK_RMS, -BACKGROUND,
                                # FILTERED, OBJECTS, -OBJECTS, SEGMENTATION,
                                # or APERTURES
# Give this in the command line instead.
#CHECKIMAGE_NAME  jabu01fvq_crclean_ext1_check.fits     # Filename for the check-image
CHECKIMAGE_NAME       ''          # Filename for the check-image

#--------------------- Memory (change with caution!) -----------------------

MEMORY_OBJSTACK  3000           # number of objects in stack
MEMORY_PIXSTACK  300000         # number of pixels in stack
MEMORY_BUFSIZE   1024           # number of lines in buffer

#----------------------------- Miscellaneous --------------------------------

VERBOSE_TYPE     NORMAL         # can be QUIET, NORMAL or FULL
WRITE_XML        N              # Write XML file (Y/N)?
XML_NAME         sex.xml        # Filename for XML output
NTHREADS          0              # Number of simultaneous threads for
```

NOTE: In Appendix F, G, and H, *updatehdr* may sometimes be set to "True"/"Yes" or "False"/"No" depending on which version of the software and which interface you are using. Also, Appendices G & H are only examples. They may not represent your last 2 **TweakReg** runs since you may conceivably try still more things to improve solutions.

Appendix F – Initial TweakReg Run Parameters:

```
_task_name_ = tweakreg#
input = *flc.fits# Input files (name, suffix, or @list)
refimage = ""# Filename of reference image
exclusions = ""# Filename for source exclusions catalogs
updatewcs = False# Update WCS keywords of input images?
writecat = False# Write out source catalogs?
clean = False# Remove intermediate files?
verbose = False# Print extra messages during processing?
runfile = tweakreg.log# Filename of processing log
expand_refcat = False# Expand reference catalog with new sources?
enforce_user_order = True# Align images in user specified order?
interactive = True# Allow interactive display of plots?

[UPDATE HEADER]
updatehdr = False# Update headers of input files with shifts?
wcsname = TWEAK# Name of updated WCS

[HEADERLET CREATION]
headerlet = False# Create headerlet for solution?
attach = True# Create headerlet FITS extension?
hdrfile = ""# Filename for headerlet FITS file
clobber = False# "Overwrite existing headerlet FITS file?"
hdrname = ""# Unique name(HDRNAME) for headerlet
author = ""# Author name for creator of headerlet
descrip = ""# Short description of headerlet solution
catalog = ""# Name of catalog used for headerlet solution
history = ""# Name of ASCII file containing history for headerlet

[OPTIONAL SHIFTFILE OUTPUT]
shiftfile = True# Create output shiftfile?
outshifts = shifts-ver1.txt# Filename of generated shiftfile
outwcs = shifts-ver1_wcs.fits# Filename of shiftfile reference WCS

[COORDINATE FILE DESCRIPTION]
catfile = coords_catfile.lis# File containing coordinate filenames for input files
xcol = 1# "Column name(s) for X positions"
ycol = 2# "Column name(s) for Y positions"
fluxcol = 3# "Column name for source flux/mag values"
fluxunits = mag# Units of flux values
xyunits = pixels# Units of X/Y positions
nbright = None# Number of brightest objects to keep
maxflux = None# Maximum flux value for valid objects
minflux = None# Minimum flux value for valid objects

[REFERENCE CATALOG DESCRIPTION]
refcat = ""# Filename of reference coordinate catalog
refxcol = 1# "Column name(s) for RA"
refycol = 2# "Column name(s) for Dec"
refxyunits = pixels# Units of sky positions
rfluxcol = 3# "Column name for source flux/mag values"
rfluxunits = mag# Units of flux values
refnbright = None# Number of brightest reference objects to keep
rmaxflux = None# Maximum flux value for valid reference objects
rminflux = None# Minimum flux value for valid reference objects

[OBJECT MATCHING PARAMETERS]
minobj = 15# Minimum number of objects acceptable for matching
searchrad = 1.0# The search radius for a match
searchunits = arcseconds# Units for search radius
use2dhist = True# Use 2d histogram to find initial offset?
see2dplot = True# See 2d histogram for initial offset?
tolerance = 1.0# Matching tolerance for xyxymatch(pixels)
separation = 0.5# Minimum object separation (pixels)
xoffset = 0.0# Initial guess for X offset(pixels)
yoffset = 0.0# Initial guess for Y offset(pixels)

[CATALOG FITTING PARAMETERS]
fitgeometry = rscale# Fitting geometry
residplot = both# Plot residuals from fit?
nclip = 3# Number of clipping iterations in fit
sigma = 3.0# Clipping limit in sigma units
ylimit = None# Upper/Lower limit (in pixels) for residplot
labelsize = 8# Font size (in points) for plot labels
```

Appendix G – Penultimate (Last Iteration) TweakReg Run Parameters:

```
_task_name_ = tweakreg#
input = *flc.fits# Input files (name, suffix, or @list)
refimage = ""# Filename of reference image
exclusions = ""# Filename for source exclusions catalogs
updatewcs = False# Update WCS keywords of input images?
writecat = True# Write out source catalogs?
clean = False# Remove intermediate files?
verbose = False# Print extra messages during processing?
runfile = tweakreg.log# Filename of processing log
expand_refcat = False# Expand reference catalog with new sources?
enforce_user_order = True# Align images in user specified order?
interactive = True# Allow interactive display of plots?

[UPDATE HEADER]
updatehdr = False# Update headers of input files with shifts?
wcsname = TWEAK# Name of updated WCS

[HEADERLET CREATION]
headerlet = False# Create headerlet for solution?
attach = True# Create headerlet FITS extension?
hdrfile = ""# Filename for headerlet FITS file
clobber = False# "Overwrite existing headerlet FITS file?"
hdrname = ""# Unique name(HDRNAME) for headerlet
author = ""# Author name for creator of headerlet
descrip = ""# Short description of headerlet solution
catalog = ""# Name of catalog used for headerlet solution
history = ""# Name of ASCII file containing history for headerlet

[OPTIONAL SHIFTFILE OUTPUT]
shiftfile = True# Create output shiftfile?
outshifts = shifts-ver3.txt# Filename of generated shiftfile
outwcs = shifts-ver3_wcs.fits# Filename of shiftfile reference WCS

[COORDINATE FILE DESCRIPTION]
catfile = coords_catfile.lis# File containing coordinate filenames for input files
xcol = 1# "Column name(s) for X positions"
ycol = 2# "Column name(s) for Y positions"
fluxcol = 3# "Column name for source flux/mag values"
fluxunits = mag# Units of flux values
xyunits = pixels# Units of X/Y positions
nbright = None# Number of brightest objects to keep
maxflux = 24.0# Maximum flux value for valid objects
minflux = 20.0# Minimum flux value for valid objects

[REFERENCE CATALOG DESCRIPTION]
refcat = ""# Filename of reference coordinate catalog
refxcol = 1# "Column name(s) for RA"
refycol = 2# "Column name(s) for Dec"
refxyunits = pixels# Units of sky positions
rfluxcol = 3# "Column name for source flux/mag values"
rfluxunits = mag# Units of flux values
refnbright = None# Number of brightest reference objects to keep
rmaxflux = None# Maximum flux value for valid reference objects
rminflux = None# Minimum flux value for valid reference objects

[OBJECT MATCHING PARAMETERS]
minobj = 15# Minimum number of objects acceptable for matching
searchrad = 1.0# The search radius for a match
searchunits = arcseconds# Units for search radius
use2dhist = True# Use 2d histogram to find initial offset?
see2dplot = True# See 2d histogram for initial offset?
tolerance = 1.0# Matching tolerance for xyxymatch(pixels)
separation = 0.5# Minimum object separation (pixels)
xoffset = 0.0# Initial guess for X offset(pixels)
yoffset = 0.0# Initial guess for Y offset(pixels)

[CATALOG FITTING PARAMETERS]
fitgeometry = rscale# Fitting geometry
residplot = both# Plot residuals from fit?
nclip = 3# Number of clipping iterations in fit
sigma = 3.0# Clipping limit in sigma units
ylimit = None# Upper/Lower limit (in pixels) for residplot
labelsize = 8# Font size (in points) for plot labels
```

Appendix H – Ultimate TweakReg Run Parameters:

```
_task_name_ = tweakreg#
input = *flc.fits# Input files (name, suffix, or @list)
refimage = ""# Filename of reference image
exclusions = ""# Filename for source exclusions catalogs
updatewcs = False# Update WCS keywords of input images?
writecat = True# Write out source catalogs?
clean = False# Remove intermediate files?
verbose = False# Print extra messages during processing?
runfile = tweakreg.log# Filename of processing log
expand_refcat = False# Expand reference catalog with new sources?
enforce_user_order = True# Align images in user specified order?
interactive = True# Allow interactive display of plots?

[UPDATE HEADER]
updatehdr = True# Update headers of input files with shifts?
wcsname = TWEAKf475w# Name of updated WCS

[HEADERLET CREATION]
headerlet = False# Create headerlet for solution?
attach = True# Create headerlet FITS extension?
hdrfile = ""# Filename for headerlet FITS file
clobber = False# "Overwrite existing headerlet FITS file?"
hdrname = ""# Unique name(HDRNAME) for headerlet
author = ""# Author name for creator of headerlet
descrip = ""# Short description of headerlet solution
catalog = ""# Name of catalog used for headerlet solution
history = ""# Name of ASCII file containing history for headerlet

[OPTIONAL SHIFTFILE OUTPUT]
shiftfile = True# Create output shiftfile?
outshifts = shifts-ver3-update.txt# Filename of generated shiftfile
outwcs = shifts-ver3-update_wcs.fits# Filename of shiftfile reference WCS

[COORDINATE FILE DESCRIPTION]
catfile = coords_catfile.lis# File containing coordinate filenames for input files
xcol = 1# "Column name(s) for X positions"
ycol = 2# "Column name(s) for Y positions"
fluxcol = 3# "Column name for source flux/mag values"
fluxunits = mag# Units of flux values
xyunits = pixels# Units of X/Y positions
nbright = None# Number of brightest objects to keep
maxflux = 24.0# Maximum flux value for valid objects
minflux = 20.0# Minimum flux value for valid objects

[REFERENCE CATALOG DESCRIPTION]
refcat = ""# Filename of reference coordinate catalog
refxcol = 1# "Column name(s) for RA"
refycol = 2# "Column name(s) for Dec"
refxyunits = pixels# Units of sky positions
rfluxcol = 3# "Column name for source flux/mag values"
rfluxunits = mag# Units of flux values
refnbright = None# Number of brightest reference objects to keep
rmaxflux = None# Maximum flux value for valid reference objects
rminflux = None# Minimum flux value for valid reference objects

[OBJECT MATCHING PARAMETERS]
minobj = 15# Minimum number of objects acceptable for matching
searchrad = 1.0# The search radius for a match
searchunits = arcseconds# Units for search radius
use2dhist = True# Use 2d histogram to find initial offset?
see2dplot = True# See 2d histogram for initial offset?
tolerance = 1.0# Matching tolerance for xyxymatch(pixels)
separation = 0.5# Minimum object separation (pixels)
xoffset = 0.0# Initial guess for X offset(pixels)
yoffset = 0.0# Initial guess for Y offset(pixels)

[CATALOG FITTING PARAMETERS]
fitgeometry = rscale# Fitting geometry
residplot = both# Plot residuals from fit?
nclip = 3# Number of clipping iterations in fit
sigma = 3.0# Clipping limit in sigma units
ylimit = None# Upper/Lower limit (in pixels) for residplot
labelsize = 8# Font size (in points) for plot labels
```